1-11 (cancelled).

12. (currently amended) A computer program device comprising:

a computer program storage device readable by a digital processing apparatus; and

a program on the program storage device and including instructions executable by the digital processing apparatus for querying at least one vertical table in a database system, the program comprising:

computer readable code means for transforming a horizontal-based SQL query into a transformed query having a format for execution against at least one vertical table.

13. (original) The computer program device of Claim 12, further comprising:

computer readable code means for defining a logical horizontal view over the vertical table;

computer readable code means for executing the transformed query against the vertical table to generate an output.

14. (original) The computer program device of Claim 13, wherein the means for transforming includes at least relational one operator.

15. (currently amended) The computer program device of Claim 14, wherein the operator receives at least one vertical table with an associated list of attribute names as input and outputs the logical horizontal table hav[r]ing column labels equal to the attribute names.

16. (original) The computer program device of Claim 15, wherein the operator is a v2h operator.

17. (original) The computer program device of Claim 15, wherein the vertical table includes object identifiers with corresponding attribute names and attribute values, and the operator executes a left outer join of a projection of object identifiers of the vertical table with a sequence of left outer joins of a set of projections of attribute values from the vertical table.

18. (original) The computer program device of Claim 12, wherein the means for transforming includes means for executing at least one projection based on the vertical table.

19. (original) The computer program device of Claim 12, wherein the means for transforming includes means for executing at least one selection from the vertical table.

20. (original) The computer program device of Claim 12, wherein the means for transforming includes means for executing at least one table join using the vertical table.

21. (currently amended) The computer program device of Claim 12, wherein the means for transforming includes means for executing at least one aggrega[h]tion.

22. (original) The computer program device of Claim 17, wherein the means for transforming includes means for executing the operator on the vertical table to render a result and then undertaking a desired set operation on the result.

23. (original) The computer program device of Claim 12, further comprising means for executing a horizontal to vertical operator against an output to transform the output to a vertical format.

24. (original) A method for extracting data from at least one vertical table in a database, comprising the acts of:

defining an enablement layer including at least a horizontal view representative of the vertical table; and

using the enablement layer, extracting data from the database based on an SQL query without requiring a user to tailor the query to a vertical format.

25. (original) The method of Claim 24, wherein the act of extracting includes:

receiving at least one SQL query against the horizontal view;

transforming the query to render a transformed query; and

executing the transformed query against the vertical table to generate an output.

26. (original) The method of Claim 25, wherein the query is transformed using at least one operator.

27. (original) The method of Claim 26, wherein the operator receives at least one vertical table with an associated list of attribute names as input and outputs the logical horizontal table having column labels equal to the attribute names.

28. (original) The method of Claim 27, wherein the operator is a v2h operator.

29. (original) The method of Claim 27, wherein the vertical table includes object identifications with corresponding attribute names and attribute values, and the operator executes a left outer join of a projection of distinct object identifiers of the vertical table with a sequence of left outer joins of a set of projections of attribute values from the vertical table.

30. (original) The method of Claim 25, wherein the transforming act includes executing at least one projection based on the vertical table.

31. (original) The method of Claim 25, wherein the transforming act includes executing at least one selection from the vertical table.

32. (original) The method of Claim 25, wherein the transforming act includes executing at least one table join using the vertical table.

33. (original) The method of Claim 25, wherein the transforming act includes executing at least one aggregation.

34. (original) The method of Claim 25, wherein the transforming act includes executing an operator on the vertical table to render a result and then undertaking a desired set operation on the result.

35. (original) The method of Claim 25, further comprising executing a horizontal to vertical operator against the output to transform the output to a vertical format.

36-38 (cancelled).

In the Office Action dated January 21, 2004 in the parent application, pending independent Claims 12 and 24 have been rejected under 35 U.S.C. §103 as being obvious over French et al. The errors in the rejection are several fold.

To summarize before explaining in more detail, French et al. structures data in vertical tables that can be represented logically by a catalogue entry, col. 4, lines 2-4, but nonetheless French et al. evidently requires a query to be formatted for execution against the vertical tables, see, for example, Claim 20 of French et al. ("receiving a user query about data values stored in a particular column"). There is thus no recognition in French et al. of transforming a query formatted for execution against a horizontal table to one suitable for execution against a vertical table, as set forth in Claim 12. Likewise, French et al. does not suggest using an enablement layer that includes a horizontal view representative of a vertical table to extract data from a database based on an SQL query without requiring a user to tailor the query to a vertical format, as set forth in Claim 24. French et al. evidently violates the recognition in the present background that "even in the hands of an expert, tailoring an SQL query for a vertical table is cumbersome and error-prone", since that is precisely what French et al. appears to do.

The previous Office Action in the parent application indeed recognizes that French et al. fails to teach receiving at least one SQL query against a logical horizontal view of a vertical table and transforming the query to render a transformed query for execution against the vertical table. However, the examiner points to col. 7, line 36 - col. 8, line 12 as a suggestion that "the query must be transformed or converted", and that "the claimed provision is inherent" and even if not inherent, the claimed transformation "would have been obvious" without any further prior art support.

This explanation misses some important points that deflate the rejection. First, in introducing the relied-upon discussion in columns 7 and 8, French et al., col. 6, lines 56-65 makes clear that the data being discussed in the relied-upon section is stored in tables of "one or more horizontal rows or records (tuples) together with vertical columns or "fields"", i.e., that the discussion which follows is a summary of how conventional horizontal tables are queried using conventional SQL querying. If this were not enough, at the end of the discussion relied on by the examiner, French et al. makes clear that the remainder of his specification (discussing vertical tables) "will focus on specific modifications" to what had just been discussed since "the previously-described needle in a haystack approach is not well-suited to DSS applications", French et al., col. 8, lines 15-21. That is, the relied-upon discussion in French et al. assumes that the reader understands that it is about how a conventional horizontal table system processes querying, as an introduction to French et al.'s subsequent divulgation of its vertical table scheme. There is nothing in the relied-upon discussion to give the reader the idea to transform the query for execution against a vertical table, since French et al. makes clear that the vertical table part of the disclosure has not yet occurred. Accordingly, nowhere in the relied-upon discussion of how conventional horizontal table query processing works nor in the subsequent divulgation of the vertical table scheme does French et al. ever indicate that a query is made against a horizontal logical view and then transformed for execution against an underlying vertical table. That idea comes only from a reading of the present specification. For this reason, the rejection is deficient.

To the extent that the examiner pursues the "inherency" theory, it is noted that the doctrine of inherency applies only to anticipation rejections, not to obviousness rejections, MPEP §2112. Moreover, to be inherent, a quality must necessarily be present in the reference, id. As the above-quoted portion of the present application makes clear, it is not necessary to transform a query against a horizontal view for execution against a vertical table if one wishes to go through the trouble of constructing the SQL query *ab initio* in a format suitable for execution against the vertical table. Thus, the allegation that French et al. "inherently" executes the claimed transformation is legally incorrect. A fair reading of French et al. gives the impression that, despite being a cumbersome chore as recognized in the present application, in French et al. queries nonetheless are structured for execution against the vertical tables.

Moreover, the examiner alleges that the table 300 of French et al. shows that French et al. "defines a logical horizontal view over the vertical table". This imputes more to French et al. than what French et al. actually teaches. There is no indication in French et al. that the table 300 is meant as anything other than an illustration of how French et al.'s vertical tables can be represented as horizontal tables, ***not*** a disclosure that French et al. actually defines such a view during operation. Nowhere does French et al. say that it does. Illustrating something for the reader is not the same thing as a teaching that the thing is actually done during operation as required by Claims 12 and 24. Once again, the examiner reads something into French et al. that only hindsight reconstruction in light of Applicant's own specification, and not a fair reading of the prior art, can render. For this further reason, the rejection is deficient.

With respect to certain dependent claims, the following arguments are germane. The allegation that French et al., col. 7, lines 41-47 teaches the operators of, e.g., Claim 14 is incorrect. As stated above, this portion of French et al. has nothing to do with vertical tables, much less the operators set forth in the present dependent claims.

Claims 20-22, 28, 29, 32, and 33 have been rejected as being obvious over French et al. in view of Graefe et al. Once again, the examiner is confusing the post-querying table pivot of Graefe et al. with a query transformation when it is alleged that it would have been obvious to incorporate Graefe et al.'s pivot operators into French et al. In accordance with the teachings of Graefe et al., the pivot operators of Graefe et al. do not do what the examiner wishes them to do in French et al. Instead, if incorporated into French et al. in accordance with what the prior art discloses, and not with what the examiner discerns in hindsight, Graefe et al. would simply pivot whatever output French et al. might produce. It would not supply an "operator" for query transformation pre-output. Again, the allegation that "the claimed provision is inherent" is simply wrong, since the "claimed provision" is not ***necessarily*** incorporated into French et al., MPEP §2112.

With this in mind, there is no prior art reason to combine Graefe et al., which simply pivots a table post-querying for presentation purposes, with French et al. which is directed to using vertical tables for actual data storage pre-querying.

In addition, Applicant makes the following observations. The Examiner makes a finding that the v2h operator of Claim 16 is the same as the pivot operation in Figure 4 of Graefe et al., without any evidentiary support. This is important, since claims are to be interpreted as one skilled in the art would interpret them, MPEP §2111.01, and there is no evidence of record that one skilled in the art would regard Figure 4 of Graefe et al. as a v2h operator that is used to transform queries as claimed, In re Dembiczak, 175 F.3D 994, 50 U.S.P.Q.2d 1614 (Fed. Cir. 1999) (the range of sources available does not diminish the requirement for **actual evidence**, and "broad conclusory statements...,standing alone, are not evidence").

Claim 17 requires that the vertical table include object identifications with corresponding attribute names and attribute values, and that the operator executes a left outer join of a projection of distinct object identifiers of the vertical table with a sequence of left outer joins of a set of projections of attribute values from the vertical table. This has been rejected on the basis of Graefe et al., Figure 4 and col. 7, lines 34-50, which indeed discusses things like outer joins ***but not in the context of being used by an operator that transforms a query***. Thus, the rejection of Claim 17 is deficient.

Claim 20, which specifies that the act of ***transforming the query*** using a table join has been rejected on the basis of Graefe et al., col. 7, lines 34-50, despite the fact that this portion of Graefe et al., like the rest of the reference, utterly fails to mention table joins in the context of transforming a query against a horizontal view for execution against an underlying vertical table. Rather, the relied-upon section of Graefe et al. is directed to extending relational calculus query expressions with relational algebraic expressions, something totally different than what is claimed.

Similarly, nothing in Graefe et al.'s teaching at col. 7, lines 47-51 teach or suggest using the mentioned "aggregations" for transforming a query against a horizontal view for execution against an underlying vertical table as recited in Claim 21.

The Examiner is cordially invited to telephone the undersigned at (619) 338-8075 for any reason which would advance the instant application to allowance.

Respectfully submitted,

John L. Rogitz
Registration No. 33,549
Attorney of Record
750 B Street, Suite 3120
San Diego, CA 92101
Telephone: (619) 338-8075

JLR:jg